

Sparse Bilinear Logistic Regression

Jianing V. Shi^{1,2*}, Yangyang Xu³, and Richard G. Baraniuk¹

¹ Department of Electrical and Computer Engineering, Rice University

² Department of Mathematics, UCLA

³ Department of Computational and Applied Mathematics, Rice University

April 17, 2014

Abstract

In this paper, we introduce the concept of sparse bilinear logistic regression for decision problems involving explanatory variables that are two-dimensional matrices. Such problems are common in computer vision, brain-computer interfaces, style/content factorization, and parallel factor analysis. The underlying optimization problem is bi-convex; we study its solution and develop an efficient algorithm based on block coordinate descent. We provide a theoretical guarantee for global convergence and estimate the asymptotical convergence rate using the Kurdyka-Łojasiewicz inequality. A range of experiments with simulated and real data demonstrate that sparse bilinear logistic regression outperforms current techniques in several important applications.

1 Introduction

Logistic regression [16] has a long history in decision problems, which are ubiquitous in computer vision [3], bioinformatics [40], gene classification [22], and neural signal processing [30]. Recently sparsity has been introduced into logistic regression to combat the curse of dimensionality in problems where only a subset of explanatory variables are informative [37]. The indices of the non-zero weights correspond to features that are informative about classification, therefore leading to feature selection. Sparse logistic regression has many attractive properties, including robustness to noise and logarithmic sample complexity bounds [29].

In the classical form of logistic regression, the explanatory variables are treated as i.i.d. vectors. However, in many real-world applications, the explanatory variables take the form of matrices. In image recognition tasks [20], for example, each feature is an image. Visual recognition tasks for video data often use a feature-based representation, such as the

*Corresponding author's email address: jianing@math.ucla.edu

scale-invariant feature transform (SIFT) [26] or histogram of oriented gradients (HOG) [7], to construct features for each frame, resulting in histogram-time feature matrices. Brain-computer interfaces based on electroencephalography (EEG) make decisions about motor action [41] using channel-time matrices.

For these and other applications, bilinear logistic regression [8] extends logistic regression to explanatory variables that take two-dimensional matrix form. The resulting dimensionality reduction of the feature space in turn yields better generalization performance. In contrast to standard logistic regression, which collapses each feature matrix into a vector and learns a single weight vector, bilinear logistic regression learns weight factors along each dimension of the matrix to form the decision boundary. It has been shown that the unregularized bilinear logistic regression outperforms linear logistic regression in several applications, including brain-computer interfaces [8]. It has also been shown that in certain visual recognition tasks, a support vector machine (SVM) applied in the bilinear feature space outperforms an SVM applied in the standard linear feature space as well as an SVM applied to a dimensionality-reduced feature space using principle component analysis (PCA) [32].

Bilinear logistic regression has also found application in style and content separation, which can improve the performance of object recognition tasks under various nuisance variables such as orientation, scale, and viewpoint [36]. Bilinear logistic regression identifies subspace projections that factor out informative features and nuisance variables, thus leading to better generalization performance.

Finally, bilinear logistic regression reveals the contributions of different dimensions to classification performance, similarly to parallel factor analysis [15]. This leads to better interpretability of the resulting decision boundary.

In this paper, we introduce sparsity to the bilinear logistic regression model and demonstrate that it improves generalization performance in a range of classification problems. Our contributions are three-fold. First, we propose a sparse bilinear regression model that fuses the key ideas behind both sparse logistic regression and bilinear logistic regression. Second, we study the properties of the solution of the bilinear logistic regression problem. Third, we develop an efficient algorithm based on block coordinate descent for solving the sparse bilinear regression problem. Both the theoretical analysis and the numerical optimization are complicated by the bi-convex nature of the problem, since the solution may become stuck at a non-stationary point. In contrast to the conventional block coordinate descent method, we solve each subproblem using the proximal method, which significantly accelerates convergence. We also provide a theoretical guarantee for global convergence and estimate the asymptotical convergence rate using a result based on the Kurdyka-Łojasiewicz inequality.

We demonstrate empirically that sparse bilinear logistic regression improves the generalization performance of the classifier under various tasks. However, due to the non-convexity

associated with the bilinear model, it remains a challenge to carry out rigorous statistical analysis using the minimax theory.

2 Sparse bilinear logistic regression

2.1 Problem Definition

We consider the following problem in this paper: Given n sample-label pairs $\{(\mathbf{X}_i, y_i)\}_{i=1}^n$, where $\mathbf{X}_i \in \mathbb{R}^{s \times t}$ is an explanatory variable in the form of a matrix and $y_i \in \{-1, +1\}$ is a categorical dependent variable, we seek a decision boundary to separate these samples.

2.2 Prior Art

2.2.1 Logistic Regression

The basic form of logistic regression [16] transforms each explanatory variable from a matrix to a vector, $\bar{\mathbf{x}}_i = \text{vec}(\mathbf{X}_i) \in \mathbb{R}^p$, where $p = st$. One seeks a hyperplane, defined as $\{\mathbf{x} : \mathbf{w}^\top \mathbf{x} + b = 0\}$, to separate these samples. For a new data sample $\bar{\mathbf{x}}_i$, its category can be predicted using a binomial model based on the margin $\mathbf{w}^\top \bar{\mathbf{x}}_i + b$. Figure 1 illustrates such an idea.

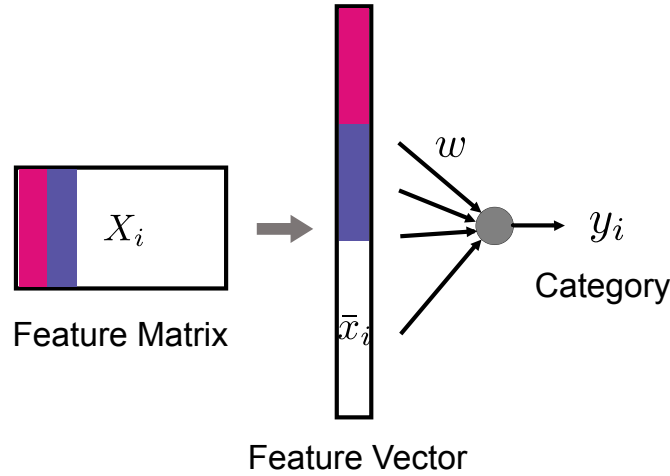


Figure 1: Illustration of logistic regression.

Essentially the logistic regression constructs a mapping from the feature vector $\bar{\mathbf{x}}_i$ to the label y_i ,

$$\Psi^{LR} : \mathbf{w}^\top \bar{\mathbf{x}}_i + b \mapsto y_i.$$

Assuming the samples of both classes are i.i.d., the conditional probability for classifier label

y_i based on sample $\bar{\mathbf{x}}_i$, according to the logistic model, takes the form of

$$p(y_i|\bar{\mathbf{x}}_i, \mathbf{w}, b) = \frac{\exp[y_i(\mathbf{w}^\top \bar{\mathbf{x}}_i + b)]}{1 + \exp[y_i(\mathbf{w}^\top \bar{\mathbf{x}}_i + b)]}, \quad i = 1, \dots, n.$$

To perform the maximum likelihood estimation (MLE) of \mathbf{w} and b , one can minimize the empirical loss function

$$\ell(\mathbf{w}, b) = \frac{1}{n} \sum_{i=1}^n \log \left(1 + \exp[-y_i(\mathbf{w}^\top \bar{\mathbf{x}}_i + b)] \right). \quad (1)$$

2.2.2 Sparse Logistic Regression

Sparse logistic regression assumes that only a subset of the decision variables are informative about the classification [37]. Typically one assumes a sparsity promoting prior on \mathbf{w} using the Laplacian prior. The maximum a posteriori (MAP) estimate for sparse logistic regression can be reduced to an ℓ_1 minimization problem

$$\min_{\mathbf{w}, b} \ell(\mathbf{w}, b) + \lambda \|\mathbf{w}\|_1, \quad (2)$$

where λ is a regularization parameter.

In the realm of machine learning, ℓ_1 regularization exists in various forms of classifiers, including ℓ_1 -regularized logistic regression [37], ℓ_1 -regularized probit regression [11, 12], ℓ_1 -regularized support vector machines [44], and ℓ_1 -regularized multinomial logistic regression [18].

The ℓ_1 -regularized logistic regression problem is convex but non-differentiable. There has been very active development on efficient numeric algorithms, including LASSO [37], G11ce [24], Grafting [31], GenLASSO [34], SCGIS [14], IRLS-LARS [9, 21], BBR [10, 13, 28], MOSEK [6], SMLR [18], interior-point method [17], FISTA [2], and HIS [35].

2.2.3 Bilinear Logistic Regression

Bilinear logistic regression was proposed in [8]. A key insight of bilinear logistic regression is to preserve the matrix structure of the explanatory variables. The decision boundary is constructed using a weight matrix \mathbf{W} , which is further factorized into $\mathbf{W} = \mathbf{U}\mathbf{V}^\top$ with two factors $\mathbf{U} \in \mathbb{R}^{s \times r}$ and $\mathbf{V} \in \mathbb{R}^{r \times t}$. Figure 2 illustrates the concept of bilinear logistic regression.

Bilinear logistic regression constructs a new mapping from the feature matrix \mathbf{X}_i to the label y_i ,

$$\Psi^{BLR} : \text{tr}(\mathbf{U}^\top \mathbf{X}_i \mathbf{V}) + b \mapsto y_i,$$

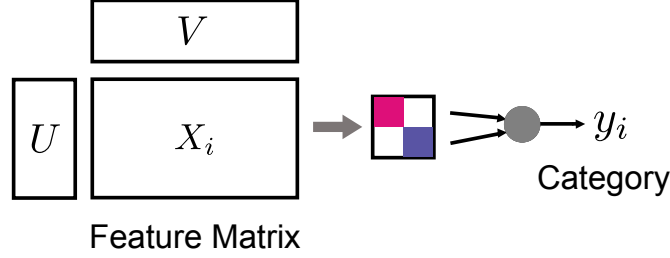


Figure 2: Illustration of bilinear logistic regression.

where $\text{tr}(\mathbf{A}) = \sum_i a_{ii}$ for a square matrix \mathbf{A} . Under these settings, the empirical loss function in (1) becomes

$$\ell(\mathbf{U}, \mathbf{V}, b) = \frac{1}{n} \sum_{i=1}^n \log \left(1 + \exp[-y_i(\text{tr}(\mathbf{U}^\top \mathbf{X}_i \mathbf{V}) + b)] \right). \quad (3)$$

The model (3) essentially identifies subspace projections that are maximally informative about classification. The variational problem generates a low-rank weight matrix $\mathbf{W} \in \mathbb{R}^{s \times t}$ that can be factorized into $\mathbf{W} \in \mathbb{R}^{s \times r}$ and $\mathbf{V} \in \mathbb{R}^{r \times t}$. One can interpret the mapping from the feature matrix to the label in the following equivalent form,

$$\Psi^{BLR} : \text{tr}(\mathbf{W} \otimes \mathbf{X}_i) + b \mapsto y_i, \quad \mathbf{W} = \mathbf{U}\mathbf{V}^\top,$$

where $\text{tr}(\mathbf{W} \otimes \mathbf{X}_i) = \sum_{j,k} (\mathbf{W})_{jk} (\mathbf{X}_i)_{jk}$.

2.3 Our New Model

2.3.1 Sparse Bilinear Logistic Regression

We introduce sparsity promoting priors on \mathbf{U} and \mathbf{V} and derive the so-called *sparse bilinear logistic regression*. The corresponding variational problem can be obtained using the MAP estimate,

$$\min_{\mathbf{U}, \mathbf{V}, b} \ell(\mathbf{U}, \mathbf{V}, b) + r_1(\mathbf{U}) + r_2(\mathbf{V}), \quad (4)$$

where r_1 and r_2 are assumed to be convex functions incorporating the priors to promote structures on \mathbf{U} and \mathbf{V} , respectively. Plugging the empirical loss function for bilinear logistic regression, the objective function of sparse bilinear logistic regression becomes

$$\min_{\mathbf{U}, \mathbf{V}, b} \frac{1}{n} \sum_{i=1}^n \log \left(1 + \exp[-y_i(\text{tr}(\mathbf{U}^\top \mathbf{X}_i \mathbf{V}) + b)] \right) + r_1(\mathbf{U}) + r_2(\mathbf{V}). \quad (5)$$

As for the sparsity promoting priors, in this paper we focus on elastic net regularization [43] of the form

$$r_1(\mathbf{U}) = \mu_1 \|\mathbf{U}\|_1 + \frac{\mu_2}{2} \|\mathbf{U}\|_F^2, \quad (6a)$$

$$r_2(\mathbf{V}) = \nu_1 \|\mathbf{V}\|_1 + \frac{\nu_2}{2} \|\mathbf{V}\|_F^2, \quad (6b)$$

where $\|\mathbf{U}\|_1 \triangleq \sum_{i,j} |u_{ij}|$. Depending on the application, other regularizers can be used. For example, one can use the total variation regularization, which we plan to explore in future work.

2.3.2 Why Sparsity?

The reasons for introducing sparsity promoting priors into bilinear logistic regression are three-fold.

First, according to [8], one limitation of bilinear logistic regression is the notorious ambiguity in the estimates. More specifically, the estimated $\hat{\mathbf{U}}$ and $\hat{\mathbf{V}}$ are subject to an arbitrary linear column space transformation

$$\text{tr}(\hat{\mathbf{U}}^\top \mathbf{X}_i \hat{\mathbf{V}}) = \text{tr}(\mathbf{G}^{-1} \mathbf{G} \hat{\mathbf{U}}^\top \mathbf{X}_i \hat{\mathbf{V}}) = \text{tr}((\hat{\mathbf{U}} \mathbf{G}^\top)^\top \mathbf{X}_i (\hat{\mathbf{V}} \mathbf{G}^{-1})), \quad (7)$$

where $\mathbf{G} \in \mathbb{R}^{r \times r}$ is an arbitrary full-rank matrix. Thus the solution to bilinear logistic regression is not unique. One can overcome such an ambiguity by introducing sparsity promoting priors on weight factors.

Second, bilinear logistic regression was originally motivated by analyzing neuroimaging data [8]. The resulting weight factors $\hat{\mathbf{U}}$, $\hat{\mathbf{V}}$ reveal spatial and temporal contributions of neural signal, with respect to certain classification tasks. Typically the neural sources generating these factors are localized spatially and temporally. Sparsity leads to feature selection, since the non-zero elements in the weight factors correspond to informative features. Therefore, it is a reasonable assumption to impose sparsity promoting priors, which can improve the interpretation of the resulting factors.

Third, sparsity improves the generalization performance of the classifier, due to its robustness to noise and logarithmic sample complexity bounds [29]. Even though the statistical analysis based on covering numbers [29] concerns linear logistic regression models, we envision that such an intuition should generalize to the bilinear model. We show empirically below that sparsity improves the generalization performance of the classifier in a range of numerical experiments.

3 Numerical Algorithm to Solve (4)

3.1 Block Coordinate Descent

We propose an efficient numerical algorithm to solve for the variational problem (4). It is based on the *block coordinate descent* method, which iteratively updates (\mathbf{U}, b) with \mathbf{V} fixed and then (\mathbf{V}, b) with \mathbf{U} fixed. The original flavor of block coordinate descent, see [27, 38] and the references therein, alternates between the following two subproblems:

$$(\mathbf{U}^k, \hat{b}^k) = \underset{(\mathbf{U}, b)}{\operatorname{argmin}} \ell(\mathbf{U}, \mathbf{V}^{k-1}, b) + r_1(\mathbf{U}), \quad (8a)$$

$$(\mathbf{V}^k, b^k) = \underset{(\mathbf{V}, b)}{\operatorname{argmin}} \ell(\mathbf{U}^k, \mathbf{V}, \hat{b}^k) + r_2(\mathbf{V}). \quad (8b)$$

The pseudocode for block coordinate descent is summarized in Algorithm 1.

Algorithm 1 Block Coordinate Descent

Input: $\{\mathbf{X}_i, y_i\}_{i=1}^n$
Initialization: Choose $(\mathbf{U}^0, \mathbf{V}^0, b^0)$
while convergence criterion not met **do**
 Compute $(\mathbf{U}^k, \hat{b}^k)$ by solving (8a)
 Compute (\mathbf{V}^k, b^k) by solving (8b)
 Let $k = k + 1$
end while

Note that even though various optimization methods exist to solve each block, due to the nonlinear form of the empirical loss function $\ell(\cdot)$, solving each block accurately can be computationally expensive.

3.2 Block Coordinate Proximal Descent

In order to accelerate computation, we solve each block using the proximal method. We call the resulting approach the *block coordinate proximal descent* method. Specifically, at iteration k , we perform the following updates:

$$\mathbf{U}^k = \underset{\mathbf{U}}{\operatorname{argmin}} \langle \nabla_{\mathbf{U}} \ell(\mathbf{U}^{k-1}, \mathbf{V}^{k-1}, b^{k-1}), \mathbf{U} - \mathbf{U}^{k-1} \rangle + \frac{L_u^k}{2} \|\mathbf{U} - \mathbf{U}^{k-1}\|_F^2 + r_1(\mathbf{U}), \quad (9a)$$

$$\hat{b}^k = \underset{b}{\operatorname{argmin}} \langle \nabla_b \ell(\mathbf{U}^{k-1}, \mathbf{V}^{k-1}, b^{k-1}), b - b^{k-1} \rangle + \frac{L_u^k}{2} (b - b^{k-1})^2, \quad (9b)$$

$$\mathbf{V}^k = \underset{\mathbf{V}}{\operatorname{argmin}} \langle \nabla_{\mathbf{V}} \ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k), \mathbf{V} - \mathbf{V}^{k-1} \rangle + \frac{L_v^k}{2} \|\mathbf{V} - \mathbf{V}^{k-1}\|_F^2 + r_2(\mathbf{V}), \quad (9c)$$

$$b^k = \underset{b}{\operatorname{argmin}} \langle \nabla_b \ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k), b - \hat{b}^k \rangle + \frac{L_v^k}{2} (b - \hat{b}^k)^2, \quad (9d)$$

where L_u^k and L_v^k are stepsize parameters to be specified in Section 3.4. Note that we have decoupled (\mathbf{U}, b) -subproblem to (9a) and (9b) since the updates of \mathbf{U} and b are independent. Similarly, (\mathbf{V}, b) -subproblem has been decoupled to (9c) and (9d).

Denote the objective function of (4) as

$$F(\mathbf{U}, \mathbf{V}, b) \triangleq \ell(\mathbf{U}, \mathbf{V}, b) + r_1(\mathbf{U}) + r_2(\mathbf{V}).$$

Let $F^k \triangleq F(\mathbf{U}^k, \mathbf{V}^k, b^k)$ and $\mathbf{W}^k \triangleq (\mathbf{U}^k, \mathbf{V}^k, b^k)$. We define *convergence criterion* as $q^k \leq \epsilon$, where

$$q^k \triangleq \max \left\{ \frac{\|\mathbf{W}^k - \mathbf{W}^{k-1}\|_F}{1 + \|\mathbf{W}^{k-1}\|_F}, \frac{|F^k - F^{k-1}|}{1 + F^{k-1}} \right\} \quad (10)$$

and $\|\mathbf{W}\|_F^2 \triangleq \|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2 + |b|^2$.

The pseudocode for block coordinate proximal descent is summarized in Algorithm 2.

Algorithm 2 Block Coordinate Proximal Descent

Input: $\{\mathbf{X}_i, y_i\}_{i=1}^n$
Initialization: Choose $(\mathbf{U}^0, \mathbf{V}^0, b^0)$
while convergence criterion not met **do**
 Compute $(\mathbf{U}^k, \hat{b}^k)$ by (9a) and (9b)
 Compute (\mathbf{V}^k, b^k) by (9c) and (9d)
 Let $k = k + 1$
end while

3.3 Solving the Subproblems

The b -subproblems (9b) and (9d) can be simply solved using gradient descent, which can be reduced to

$$\hat{b}^k = b^{k-1} - \frac{1}{L_u^k} \nabla_b \ell(\mathbf{U}^{k-1}, \mathbf{V}^{k-1}, b^{k-1}), \quad (11a)$$

$$b^k = \hat{b}^k - \frac{1}{L_v^k} \nabla_b \ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k). \quad (11b)$$

The \mathbf{U} -subproblem (9a) and \mathbf{V} -subproblem (9c) are both strongly convex and can be solved by various convex programming solvers. Since the dimension of input data can be large, it is important to solve the subproblems very efficiently. The beauty of using the proximal method is its admission for closed-form solutions. More specifically, for elastic net regularization terms r_1 and r_2 defined as (6), both (9a) and (9c) admits closed form

solutions

$$\mathbf{U}^k = \mathcal{S}_{\tau_u} \left(\frac{L_u^k \mathbf{U}^{k-1} - \nabla_{\mathbf{U}} \ell(\mathbf{U}^{k-1}, \mathbf{V}^{k-1}, b^{k-1})}{L_u^k + \mu_2} \right), \quad (12a)$$

$$\mathbf{V}^k = \mathcal{S}_{\tau_v} \left(\frac{L_v^k \mathbf{V}^{k-1} - \nabla_{\mathbf{V}} \ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k)}{L_v^k + \nu_2} \right), \quad (12b)$$

where $\tau_u = \frac{\mu_1}{L_u^k + \mu_2}$, $\tau_v = \frac{\nu_1}{L_v^k + \nu_2}$, and $\mathcal{S}_\tau(\cdot)$ is the component-wise shrinkage defined as

$$(\mathcal{S}_\tau(\mathbf{Z}))_{ij} = \begin{cases} z_{ij} - \tau, & \text{if } z_{ij} > \tau; \\ z_{ij} + \tau, & \text{if } z_{ij} < -\tau; \\ 0, & \text{if } |z_{ij}| \leq \tau. \end{cases}$$

The proximal method leads to closed-form solution for each subproblem, and the entire algorithm only involves matrix-vector multiplication and component-wise shrinkage operator. Therefore our numerical algorithm will be computationally efficient. We will corroborate this statement using numerical experiments.

3.4 Selection of L_u^k and L_v^k

To ensure the sequence generated by Algorithm 2 attains sufficient decrease in the objective function, L_u^k is typically chosen as a Lipschitz constant of $\nabla_{(\mathbf{U}, b)} \ell(\mathbf{U}, \mathbf{V}^{k-1}, b)$ with respect to (\mathbf{U}, b) . More precisely, for all (\mathbf{U}, b) and $(\tilde{\mathbf{U}}, \tilde{b})$, we have

$$\|\nabla_{(\mathbf{U}, b)} \ell(\mathbf{U}, \mathbf{V}^{k-1}, b) - \nabla_{(\mathbf{U}, b)} \ell(\tilde{\mathbf{U}}, \mathbf{V}^{k-1}, \tilde{b})\|_F \leq L_u^k \|(\mathbf{U}, b) - (\tilde{\mathbf{U}}, \tilde{b})\|_F, \quad (13)$$

where $\|(\mathbf{U}, b)\|_F := \sqrt{\|\mathbf{U}\|_F^2 + b^2}$. Similarly, L_v^k can be chosen as a Lipschitz constant of $\nabla_{(\mathbf{V}, b)} \ell(\mathbf{U}^k, \mathbf{V}, b)$ with respect to (\mathbf{V}, b) . The next lemma shows that the two partial gradients $\nabla_{(\mathbf{U}, b)} \ell(\mathbf{U}, \mathbf{V}, b)$ and $\nabla_{(\mathbf{V}, b)} \ell(\mathbf{U}, \mathbf{V}, b)$ are Lipschitz continuous with constants dependent on \mathbf{U} and \mathbf{V} respectively.

Lemma 3.1 *The partial gradients $\nabla_{(\mathbf{U}, b)} \ell(\mathbf{U}, \mathbf{V}, b)$ and $\nabla_{(\mathbf{V}, b)} \ell(\mathbf{U}, \mathbf{V}, b)$ are Lipschitz continuous with constants*

$$L_u = \frac{\sqrt{2}}{n} \sum_{i=1}^n (\|\mathbf{X}_i \mathbf{V}\|_F + 1)^2, \quad (14a)$$

$$L_v = \frac{\sqrt{2}}{n} \sum_{i=1}^n (\|\mathbf{X}_i^\top \mathbf{U}\|_F + 1)^2, \quad (14b)$$

Proof. By straightforward calculation, we have

$$\nabla_{\mathbf{U}}\ell(\mathbf{U}, \mathbf{V}, b) = -\frac{1}{n} \sum_{i=1}^n \left(1 + \exp[y_i(\text{tr}(\mathbf{U}^\top \mathbf{X}_i \mathbf{V}) + b)]\right)^{-1} y_i \mathbf{X}_i \mathbf{V}, \quad (15a)$$

$$\nabla_{\mathbf{V}}\ell(\mathbf{U}, \mathbf{V}, b) = -\frac{1}{n} \sum_{i=1}^n \left(1 + \exp[y_i(\text{tr}(\mathbf{U}^\top \mathbf{X}_i \mathbf{V}) + b)]\right)^{-1} y_i \mathbf{X}_i^\top \mathbf{U}, \quad (15b)$$

$$\nabla_b\ell(\mathbf{U}, \mathbf{V}, b) = -\frac{1}{n} \sum_{i=1}^n \left(1 + \exp[y_i(\text{tr}(\mathbf{U}^\top \mathbf{X}_i \mathbf{V}) + b)]\right)^{-1} y_i. \quad (15c)$$

For any (\mathbf{U}, b) and $(\tilde{\mathbf{U}}, \tilde{b})$, we have

$$\begin{aligned} & \|\nabla_{(\mathbf{U}, b)}\ell(\mathbf{U}, \mathbf{V}, b) - \nabla_{(\tilde{\mathbf{U}}, \tilde{b})}\ell(\tilde{\mathbf{U}}, \mathbf{V}, \tilde{b})\|_F \\ & \leq \frac{1}{n} \sum_{i=1}^n \left| \left(1 + \exp[y_i(\text{tr}(\mathbf{U}^\top \mathbf{X}_i \mathbf{V}) + b)]\right)^{-1} - \left(1 + \exp[y_i(\text{tr}(\tilde{\mathbf{U}}^\top \mathbf{X}_i \mathbf{V}) + \tilde{b})]\right)^{-1} \right| \\ & \quad (\|\mathbf{X}_i \mathbf{V}\|_F + 1) \\ & \leq \frac{1}{n} \sum_{i=1}^n \left(\|\mathbf{U} - \tilde{\mathbf{U}}\|_F \|\mathbf{X}_i \mathbf{V}\|_F + |b - \tilde{b}| \right) (\|\mathbf{X}_i \mathbf{V}\|_F + 1) \\ & \leq \frac{1}{n} \sum_{i=1}^n (\|\mathbf{X}_i \mathbf{V}\|_F + 1)^2 \left(\|\mathbf{U} - \tilde{\mathbf{U}}\|_F + |b - \tilde{b}| \right) \\ & \leq \frac{\sqrt{2}}{n} \sum_{i=1}^n (\|\mathbf{X}_i \mathbf{V}\|_F + 1)^2 \|(\mathbf{U}, b) - (\tilde{\mathbf{U}}, \tilde{b})\|_F, \end{aligned}$$

where in the third inequality we have used the inequality

$$|(1 + e^s)^{-1} - (1 + e^q)^{-1}| \leq |s - q|$$

and the last inequality follows from

$$\|\mathbf{U} - \tilde{\mathbf{U}}\|_F + |b - \tilde{b}| \leq \sqrt{2} \|(\mathbf{U}, b) - (\tilde{\mathbf{U}}, \tilde{b})\|_F$$

by the Cauchy-Schwarz inequality. This completes the proof of (14a), and (14b) can be shown in the same way. ■

However, L_u^k and L_v^k chosen in such a manner may be too large, slowing convergence. Therefore we have chosen to use an alternative and efficient way to dynamically update them. Specifically, we let

$$L_u^k = \max(L_{\min}, L_u^{k-1} \eta^{n_u^k}) \quad (16)$$

where $L_{\min} > 0$, $\eta > 1$, and $n_u^k \geq -1$ is the smallest integer such that

$$\begin{aligned}
& \ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k) \\
& \leq \ell(\mathbf{U}^{k-1}, \mathbf{V}^{k-1}, b^{k-1}) \\
& \quad + \langle \nabla_{\mathbf{U}} \ell(\mathbf{U}^{k-1}, \mathbf{V}^{k-1}, b^{k-1}), \mathbf{U}^k - \mathbf{U}^{k-1} \rangle + \langle \nabla_b \ell(\mathbf{U}^{k-1}, \mathbf{V}^{k-1}, b^{k-1}), \hat{b}^k - b^{k-1} \rangle \\
& \quad + \frac{L_u^k}{2} \|\mathbf{U}^k - \mathbf{U}^{k-1}\|_F^2 + \frac{L_u^k}{2} (\hat{b}^k - b^{k-1})^2,
\end{aligned} \tag{17}$$

and let

$$L_v^k = \max(L_{\min}, L_v^{k-1} \eta^{n_v^k}), \tag{18}$$

where $n_v^k \geq -1$ is the smallest integer such that

$$\begin{aligned}
& \ell(\mathbf{U}^k, \mathbf{V}^k, b^k) \\
& \leq \ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k) \\
& \quad + \langle \nabla_{\mathbf{V}} \ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k), \mathbf{V}^k - \mathbf{V}^{k-1} \rangle + \langle \nabla_b \ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k), b^k - \hat{b}^k \rangle \\
& \quad + \frac{L_v^k}{2} \|\mathbf{V}^k - \mathbf{V}^{k-1}\|_F^2 + \frac{L_v^k}{2} (b^k - \hat{b}^k)^2.
\end{aligned} \tag{19}$$

The inequalities (17) and (19) guarantee sufficient decrease of the objective and are required for convergence. If L_u^k and L_v^k are taken as Lipschitz constants of $\nabla_{(\mathbf{U}, b)} \ell(\mathbf{U}, \mathbf{V}^{k-1}, b)$ and $\nabla_{(\mathbf{V}, b)} \ell(\mathbf{U}^k, \mathbf{V}, b)$, then the two inequalities must hold. In our dynamical updating rule, note that in (16) and (18), we allow n_u^k and n_v^k to be negative, namely, L_u^k and L_v^k can be smaller than their previous values. Moreover, n_u^k and n_v^k must be finite if the sequence $\{(\mathbf{U}^k, \mathbf{V}^k)\}$ is bounded, and thus the updates in (16) and (18) are well-defined.

4 Convergence Analysis

We now establish the global convergence of the block coordinate proximal descent algorithm for sparse bilinear logistic regression, as well as estimate its asymptotic convergence rate.

Our analysis mainly follows [42], which establishes global convergence of the *cyclic* block coordinate proximal method assuming the Kurdyka-Łojasiewicz inequality (see Definition 4.1 below). Since our algorithm updates b -block twice during each iteration, its convergence result cannot be obtained directly from [42]. The work [39] also establishes global convergence results with rate estimation for the block coordinate proximal method. However, it assumes the so-called *local Lipschitzian error bound*, which is not known to hold for our problem. Throughout our analysis, we make the following assumption.

Assumption 4.1 *We assume the objective function F is lower bounded and the problem (4) has at least one stationary point. In addition, we assume the sequence $\{\mathbf{W}^k\}$ is bounded.*

Remark 4.1 According to (14), L_u^k, L_v^k must be bounded if $\{\mathbf{W}^k\}$ is bounded. In addition, for the regularization terms, r_1 set by (6a) and r_2 taken as (6b), then F is lower bounded by zero, and (4) has at least one solution.

Theorem 4.1 (Subsequence Convergence) Under Assumption 4.1, let $\{\mathbf{W}^k\}$ be the sequence generated from Algorithm 2. Then any limit point $\bar{\mathbf{W}}$ of $\{\mathbf{W}^k\}$ is a stationary point of (4).

Proof. From Lemma 2.3 of [2], we have

$$F(\mathbf{W}^{k-1}) - F(\mathbf{U}^k, \hat{b}^k, \mathbf{V}^{k-1}) \geq \frac{L_u^k}{2} (\|\mathbf{U}^{k-1} - \mathbf{U}^k\|_F^2 + |b^{k-1} - \hat{b}^k|^2),$$

and

$$F(\mathbf{U}^k, \hat{b}^k, \mathbf{V}^{k-1}) - F(\mathbf{W}^k) \geq \frac{L_v^k}{2} (\|\mathbf{V}^{k-1} - \mathbf{V}^k\|_F^2 + |\hat{b}^k - b^k|^2).$$

Assume $\min(L_u^k, L_v^k) \geq L_{\min}$ for all k . Summing up the above two inequality gives

$$F(\mathbf{W}^{k-1}) - F(\mathbf{W}^k) \geq \frac{L_{\min}}{2} (\|\mathbf{U}^{k-1} - \mathbf{U}^k\|_F^2 + \|\mathbf{V}^{k-1} - \mathbf{V}^k\|_F^2 + |b^{k-1} - \hat{b}^k|^2 + |\hat{b}^k - b^k|^2), \quad (20)$$

which yields

$$F(\mathbf{W}^0) - F(\mathbf{W}^N) \geq \sum_{k=1}^N (\|\mathbf{U}^{k-1} - \mathbf{U}^k\|_F^2 + \|\mathbf{V}^{k-1} - \mathbf{V}^k\|_F^2 + |b^{k-1} - \hat{b}^k|^2 + |\hat{b}^k - b^k|^2).$$

Letting $N \rightarrow \infty$ and observing $F \geq 0$, we have

$$\sum_{k=1}^{\infty} (\|\mathbf{U}^{k-1} - \mathbf{U}^k\|_F^2 + \|\mathbf{V}^{k-1} - \mathbf{V}^k\|_F^2 + |b^{k-1} - \hat{b}^k|^2 + |\hat{b}^k - b^k|^2) \leq \infty.$$

Hence, $\mathbf{W}^k - \mathbf{W}^{k-1} \rightarrow \mathbf{0}$.

Let $\bar{\mathbf{W}}$ be a limit point. Hence, there exists a subsequence $\{\mathbf{W}^k\}_{k \in \mathcal{K}}$ converging to $\bar{\mathbf{W}}$. Passing to another subsequence, we can assume that $\{L_u^k\}_{k \in \mathcal{K}}$ and $\{L_v^k\}_{k \in \mathcal{K}}$ converge to \bar{L}_u and \bar{L}_v respectively. Note that $\{\mathbf{W}^{k-1}\}_{k \in \mathcal{K}}$ also converges to $\bar{\mathbf{W}}$ and $\{\hat{b}^k\}_{k \in \mathcal{K}} \rightarrow \bar{b}$. Letting $k \in \mathcal{K}$ and $k \rightarrow \infty$ in (9a), we have

$$\bar{\mathbf{U}} = \underset{\mathbf{U}}{\operatorname{argmin}} \langle \nabla_{\mathbf{U}} \ell(\bar{\mathbf{U}}, \bar{\mathbf{V}}, \bar{b}), \mathbf{U} - \bar{\mathbf{U}} \rangle + \frac{\bar{L}_u}{2} \|\mathbf{U} - \bar{\mathbf{U}}\|_F^2 + r_1(\mathbf{U}),$$

which implies $\mathbf{0} \in \nabla_{\mathbf{U}} \ell(\bar{\mathbf{U}}, \bar{\mathbf{V}}, \bar{b}) + \partial r_1(\bar{\mathbf{U}})$. Similarly, one can show $\mathbf{0} \in \nabla_{\mathbf{V}} \ell(\bar{\mathbf{U}}, \bar{\mathbf{V}}, \bar{b}) + \partial r_2(\bar{\mathbf{V}})$ and $\nabla_b \ell(\bar{\mathbf{U}}, \bar{\mathbf{V}}, \bar{b}) = 0$. Hence, $\bar{\mathbf{W}}$ is a critical point. ■

In order to establish global convergence, we utilize Kurdyka-Łojasiewicz inequality defined below [5, 19, 23].

Definition 4.1 (Kurdyka-Łojasiewicz Inequality) A function F is said to satisfy the Kurdyka-Łojasiewicz inequality at point $\bar{\mathbf{W}}$, if there exists $\theta \in [0, 1)$ such that

$$\frac{|F(\mathbf{W}) - F(\bar{\mathbf{W}})|^\theta}{\text{dist}(\mathbf{0}, \partial F(\mathbf{W}))} \quad (21)$$

is bounded for any \mathbf{W} near $\bar{\mathbf{W}}$, where $\partial F(\mathbf{W})$ is the limiting subdifferential [33] of F at \mathbf{W} , and $\text{dist}(\mathbf{0}, \partial F(\mathbf{W})) \triangleq \min\{\|\mathbf{Y}\|_F : \mathbf{Y} \in \partial F(\mathbf{W})\}$.

Theorem 4.2 (Global Convergence) Suppose Assumption 4.1 holds and F satisfies the Kurdyka-Łojasiewicz inequality at a limit point $\bar{\mathbf{W}}$ of $\{\mathbf{W}^k\}$, then \mathbf{W}^k converges to $\bar{\mathbf{W}}$.

Proof. The boundedness of $\{\mathbf{W}^k\}$ implies that all intermediate points are bounded. Hence, there exists a constant L_{\max} such that $L_u^k, L_v^k \leq L_{\max}$ for all k , and also there is a constant L_G such that for all k

$$\|\nabla_{\mathbf{U}}\ell(\mathbf{W}^k) - \nabla_{\mathbf{U}}\ell(\mathbf{W}^{k-1})\|_F \leq L_G \|\mathbf{W}^k - \mathbf{W}^{k-1}\|_F, \quad (22a)$$

$$\|\nabla_{\mathbf{V}}\ell(\mathbf{W}^k) - \nabla_{\mathbf{V}}\ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k)\|_F \leq L_G \|\mathbf{W}^k - (\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k)\|_F, \quad (22b)$$

$$\|\nabla_b\ell(\mathbf{W}^k) - \nabla_b\ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k)\|_F \leq L_G \|\mathbf{W}^k - (\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k)\|_F. \quad (22c)$$

Let $\bar{\mathbf{W}}$ be a limit point of $\{\mathbf{W}^k\}$ and assume F satisfies KL-inequality within $\mathbb{B}_\rho(\bar{\mathbf{W}}) \triangleq \{\mathbf{W} : \|\mathbf{W} - \bar{\mathbf{W}}\|_F \leq \rho\}$, namely, there exists constants $0 \leq \theta < 1$ and $C > 0$ such that

$$\frac{|F(\mathbf{W}) - F(\bar{\mathbf{W}})|^\theta}{\text{dist}(\mathbf{0}, \partial F(\mathbf{W}))} \leq C, \quad \forall \mathbf{W} \in \mathbb{B}_\rho(\bar{\mathbf{W}}). \quad (23)$$

Noting $\mathbf{W}^k - \mathbf{W}^{k-1} \rightarrow \mathbf{0}$, $|b^k - \hat{b}^k| \rightarrow 0$, and the continuity of $\phi(s) = s^{1-\theta}$, we can take sufficiently large k_0 such that

$$2\|\mathbf{W}^{k_0} - \mathbf{W}^{k_0+1}\|_F + \|\bar{\mathbf{W}} - \mathbf{W}^{k_0}\|_F + |b^{k_0+1} - \hat{b}^{k_0+1}| + \frac{1}{\tilde{C}^2} \phi(F(\mathbf{W}^{k_0}) - F(\bar{\mathbf{W}})) \leq \rho, \quad (24)$$

where $\tilde{C} = \sqrt{\frac{(1-\theta)L_{\min}}{8C \cdot (3L_G + 2L_{\max})}}$. Without loss of generality, we assume $k_0 = 0$ (i.e., take \mathbf{W}^{k_0} as starting point), since the convergence of $\{\mathbf{W}^k\}_{k \geq 0}$ is equivalent to that of $\{\mathbf{W}^k\}_{k \geq k_0}$. In addition, we denote $F_k = F(\mathbf{W}^k) - F(\bar{\mathbf{W}})$ and note $F_k \geq 0$ from the non-increasing monotonicity of $\{F(\mathbf{W}^k)\}$.

From (9), we have

$$-\nabla_{\mathbf{U}}\ell(\mathbf{W}^{k-1}) + \nabla_{\mathbf{U}}\ell(\mathbf{W}^k) - L_u^k(\mathbf{U}^k - \mathbf{U}^{k-1}) \in \partial r_1(\mathbf{U}^k) + \nabla_{\mathbf{U}}\ell(\mathbf{W}^k), \quad (25a)$$

$$-\nabla_{\mathbf{V}}\ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k) + \nabla_{\mathbf{V}}\ell(\mathbf{W}^k) - L_v^k(\mathbf{V}^k - \mathbf{V}^{k-1}) \in \partial r_2(\mathbf{V}^k) + \nabla_{\mathbf{V}}\ell(\mathbf{W}^k), \quad (25b)$$

$$-\nabla_b\ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k) + \nabla_b\ell(\mathbf{W}^k) - L_v^k(b^k - \hat{b}^k) = \nabla_b\ell(\mathbf{W}^k). \quad (25c)$$

Hence,

$$\begin{aligned}
& \text{dist}(\mathbf{0}, \partial F(\mathbf{W}^k)) \\
& \leq \|\nabla_{\mathbf{U}} \ell(\mathbf{W}^k) - \nabla_{\mathbf{U}} \ell(\mathbf{W}^{k-1})\|_F + L_u^k \|\mathbf{U}^k - \mathbf{U}^{k-1}\|_F + \|\nabla_{\mathbf{V}} \ell(\mathbf{W}^k) - \nabla_{\mathbf{V}} \ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k)\|_F \\
& \quad + L_v^k \|\mathbf{V}^k - \mathbf{V}^{k-1}\|_F + \|\nabla_b \ell(\mathbf{W}^k) - \nabla_b \ell(\mathbf{U}^k, \mathbf{V}^{k-1}, \hat{b}^k)\|_F + L_v^k |b^k - \hat{b}^k| \\
& \leq (3L_G + 2L_{\max})(\|\mathbf{W}^k - \mathbf{W}^{k-1}\|_F + |b^k - \hat{b}^k|). \tag{26}
\end{aligned}$$

Note that (20) implies

$$F_k - F_{k+1} \geq \frac{L_{\min}}{4} (\|\mathbf{W}^{k+1} - \mathbf{W}^k\|_F^2 + |b^{k+1} - \hat{b}^{k+1}|^2).$$

Assume $\mathbf{W}^k \in \mathbb{B}_\rho(\bar{\mathbf{W}})$ for $0 \leq k \leq N$. We go to show $\mathbf{W}^{N+1} \in \mathbb{B}_\rho(\bar{\mathbf{W}})$. By the concavity of $\phi(s) = s^{1-\theta}$ and KL-inequality (23), we have

$$\phi(F_k) - \phi(F_{k+1}) \geq \phi'(F_k)(F_k - F_{k+1}) \geq \frac{(1-\theta)L_{\min}(\|\mathbf{W}^{k+1} - \mathbf{W}^k\|_F^2 + |b^{k+1} - \hat{b}^{k+1}|^2)}{4C \cdot (3L_G + 2L_{\max})(\|\mathbf{W}^k - \mathbf{W}^{k-1}\|_F + |b^k - \hat{b}^k|)}, \tag{27}$$

which together with Cauchy-Schwarz inequality gives

$$\tilde{C}(\|\mathbf{W}^k - \mathbf{W}^{k+1}\|_F + |b^{k+1} - \hat{b}^{k+1}|) \leq \frac{\tilde{C}}{2} (\|\mathbf{W}^{k-1} - \mathbf{W}^k\|_F + |b^k - \hat{b}^k|) + \frac{1}{2\tilde{C}} (\phi(F_k) - \phi(F_{k+1})). \tag{28}$$

Summing up the above inequality gives

$$\frac{\tilde{C}}{2} \sum_{k=1}^N (\|\mathbf{W}^k - \mathbf{W}^{k+1}\|_F + |b^{k+1} - \hat{b}^{k+1}|) \leq \frac{\tilde{C}}{2} (\|\mathbf{W}^0 - \mathbf{W}^1\|_F + |b^1 - \hat{b}^1|) + \frac{1}{2\tilde{C}} (\phi(F_0) - \phi(F_{N+1})). \tag{29}$$

Hence,

$$\begin{aligned}
& \|\mathbf{W}^{N+1} - \bar{\mathbf{W}}\|_F \\
& \leq \sum_{k=1}^N \|\mathbf{W}^k - \mathbf{W}^{k+1}\|_F + \|\mathbf{W}^0 - \mathbf{W}^1\|_F + \|\bar{\mathbf{W}} - \mathbf{W}^0\|_F \\
& \leq 2\|\mathbf{W}^0 - \mathbf{W}^1\|_F + \|\bar{\mathbf{W}} - \mathbf{W}^0\|_F + |b^1 - \hat{b}^1| + \frac{1}{\tilde{C}^2} \phi(F_0) \leq \rho, \tag{30}
\end{aligned}$$

where the last inequality is from (24). Hence, $\mathbf{W}^{N+1} \in \mathbb{B}_\rho(\bar{\mathbf{W}})$, and by induction, $\mathbf{W}^k \in \mathbb{B}_\rho(\bar{\mathbf{W}})$ for all k . Therefore, (29) holds for all N . Letting $N \rightarrow \infty$ in (29) yields

$$\sum_{k=1}^{\infty} \|\mathbf{W}^k - \mathbf{W}^{k+1}\|_F < \infty.$$

Therefore $\{\mathbf{W}^k\}$ is a Cauchy sequence and thus converges to the limit point $\bar{\mathbf{W}}$. ■

Remark 4.2 Note that the logistic function ℓ is real analytic. If r_1 and r_2 are taken as in (6), then they are semi-algebraic functions [4], and, according to [42], F satisfies the Kurdyka-Łojasiewicz inequality at every point.

Theorem 4.3 (Convergence Rate) Depending on θ in (21), we have the following convergence rates:

1. If $\theta = 0$, then \mathbf{W}^k converges to $\bar{\mathbf{W}}$ in finite iterations;
2. If $\theta \in (0, \frac{1}{2}]$, then \mathbf{W}^k converges to $\bar{\mathbf{W}}$ at least linearly, i.e., $\|\mathbf{W}^k - \bar{\mathbf{W}}\|_F \leq C\tau^k$ for some positive constants C and $\tau < 1$;
3. If $\theta \in (\frac{1}{2}, 1)$, then \mathbf{W}^k converges to $\bar{\mathbf{W}}$ at least sublinearly. Specifically, $\|\mathbf{W}^k - \bar{\mathbf{W}}\|_F \leq Ck^{-\frac{1-\theta}{2\theta-1}}$ for some constant $C > 0$.

Proof. We estimate the convergence rates for different θ in (23).

Case 1: $\theta = 0$. We claim \mathbf{W}^k converges to $\bar{\mathbf{W}}$ in finite iterations, i.e., there is k_0 such that $\mathbf{W}^k = \bar{\mathbf{W}}$ for all $k \geq k_0$. Otherwise, $F(\mathbf{W}^k) > F(\bar{\mathbf{W}})$ for all k since if $F(\mathbf{W}^{k_0}) = F(\bar{\mathbf{W}})$ then $\mathbf{W}^k = \bar{\mathbf{W}}$ for all $k \geq k_0$. By KL-inequality (23), we have $C \cdot \text{dist}(\mathbf{0}, \partial F(\mathbf{W}^k)) \geq 1$ for all k . However, (25) indicates $\text{dist}(\mathbf{0}, \partial F(\mathbf{W}^k)) \rightarrow 0$ as $k \rightarrow \infty$. Therefore, if $\theta = 0$, then \mathbf{W}^k converges to $\bar{\mathbf{W}}$ in finite iterations.

Case 2: $\theta \in (0, \frac{1}{2}]$. Denote $S_N = \sum_{k=N}^{\infty} (\|\mathbf{W}^k - \mathbf{W}^{k+1}\|_F + |b^{k+1} - \hat{b}^{k+1}|)$. Note that (28) holds for all k . Summing (28) over k gives $S_N \leq S_{N-1} - S_N + \frac{1}{2\hat{C}^2} F_N^{1-\theta}$. By (23) and (26), we have

$$F_N^{1-\theta} = (F_N^\theta)^{\frac{1-\theta}{\theta}} \leq (C \cdot (3L_G + 2L_{\max}))^{\frac{1-\theta}{\theta}} (S_{N-1} - S_N)^{\frac{1-\theta}{\theta}}.$$

Hence,

$$S_N \leq S_{N-1} - S_N + \hat{C}(S_{N-1} - S_N)^{\frac{1-\theta}{\theta}}, \quad (31)$$

where $\hat{C} = \frac{1}{2\hat{C}^2} (C \cdot (3L_G + 2L_{\max}))^{\frac{1-\theta}{\theta}}$. Note that $S_{N-1} - S_N \leq 1$ as N is sufficiently large, and also $\frac{1-\theta}{\theta} \geq 1$ when $\theta \in (0, \frac{1}{2}]$. Therefore, $(S_{N-1} - S_N)^{\frac{1-\theta}{\theta}} \leq S_{N-1} - S_N$, and thus (31) implies $S_N \leq (1 + \hat{C})(S_{N-1} - S_N)$. Hence, $S_N \leq \frac{1+\hat{C}}{2+\hat{C}} S_{N-1} \leq (\frac{1+\hat{C}}{2+\hat{C}})^N S_0$. Noting that $\|\mathbf{W}^N - \bar{\mathbf{W}}\|_F \leq S_N$, we have

$$\|\mathbf{W}^N - \bar{\mathbf{W}}\|_F \leq \left(\frac{1+\hat{C}}{2+\hat{C}}\right)^N S_0.$$

Case 3: $\theta \in (\frac{1}{2}, 1)$. Note $\frac{1-\theta}{\theta} < 1$. Hence, (31) implies that

$$S_N \leq (1 + \hat{C})(S_{N-1} - S_N)^{\frac{1-\theta}{\theta}}.$$

Through the same argument in the proof of Theorem 2 of [1], we can show

$$S_N \leq c \cdot N^{-\frac{1-\theta}{2\theta-1}},$$

for some constant c . This completes the proof. ■

Remark 4.3 *Note that the value of θ depends not only on F but also on $\bar{\mathbf{W}}$. The paper [42] gives estimates for different classes of functions. Since the limit point is not known ahead, we cannot estimate θ . However, our numerical results in Section 5 indicate that our algorithm converges asymptotically superlinearly and thus θ should be less than $\frac{1}{2}$ for our tests.*

5 Numerical Results

5.1 Implementation

Since the variational problem (4) is non-convex, the starting point is significant for both the solution quality and convergence speed of our algorithms. Throughout our tests, we simply set $b^0 = 0$ and chose $(\mathbf{U}^0, \mathbf{V}^0)$ as follows.

Let $\mathbf{X}^{av} = \frac{1}{n} \sum_{i=1}^n \mathbf{X}_i$. Then set \mathbf{U}^0 to the negative of the first r left singular vectors and \mathbf{V}^0 to the first r right singular vectors of \mathbf{X}^{av} corresponding to its first r largest singular values.

The intuition of choosing such $(\mathbf{U}^0, \mathbf{V}^0)$ is that it is one minimizer of $\frac{1}{n} \sum_{i=1}^n \text{tr}(\mathbf{U}^\top \mathbf{X}_i \mathbf{V})$, which is exactly the first-order Taylor expansion of $\ell(\mathbf{U}, \mathbf{V}, 0)$ at the origin, under constraints $\mathbf{U}^\top \mathbf{U} = \mathbf{I}$ and $\mathbf{V}^\top \mathbf{V} = \mathbf{I}$. Unless specified, the algorithms were terminated if they ran over 500 iterations or the relative error $q^k \leq 10^{-3}$.

5.2 Scalability

In order to demonstrate the computational benefit of the proximal method, we compared Algorithm 2 with Algorithm 1 on randomly generated data. Each data point¹ in class “+1” was generated by MATLAB command `randn(s,t)+1` and each one in class “-1” by `randn(s,t)-1`. The sample size was fixed to $n = 100$, and the dimensions were kept by $s = t$ with s varying among $\{50, 100, 250, 500, 750, 1000\}$. We tested two sets of parameters for the scalability test. We ran each algorithm with one set of parameters for 5 times with different random data.

Table 1 shows the average running time and the median number of iterations. From the table, we see that both Algorithm 1 and Algorithm 2 are scalable to large-scale dataset and converge within the given tolerance after quite a few iterations. The per-iteration running

¹We use synthetic data simply for scalability and speed test. For other numerical experiments, we use real-world datasets.

Table 1: Scalability and comparison of Algorithms 1 and 2. Shown are the average running time and median number of iterations.

	Algorithm 1		Algorithm 2	
$\mu_1 = \nu_1 = 0.1, \mu_2 = \nu_2 = 1$				
(s, t)	time (sec.)	iter	time (sec.)	iter
(50, 50)	0.79	5	0.03	9
(100, 100)	1.13	6	0.06	11
(250, 250)	3.89	6	0.56	31
(500, 500)	9.96	5	1.80	4
(750, 750)	18.60	7	4.04	4
(1000, 1000)	16.25	3	7.92	4
$\mu_1 = \nu_1 = 0.1, \mu_2 = \nu_2 = 0$				
(s, t)	time (sec.)	iter	time (sec.)	iter
(50, 50)	6.87	17	0.37	282
(100, 100)	14.39	29	0.38	47
(250, 250)	21.73	8	3.49	28
(500, 500)	78.32	7	4.07	11
(750, 750)	129.23	8	4.31	4
(1000, 1000)	218.49	9	8.19	4

time increases almost linearly with respect to the data size. In addition, Algorithm 2 is much faster than Algorithm 1 in terms of running time. Note the degree of speedup depends on the parameters. In the first experiment, where ℓ_2 regularization dominates ($\mu_1 = \nu_1 = 0.1, \mu_2 = \nu_2 = 1$), Algorithm 2 is twice as fast as Algorithm 1. In the second experiment, where ℓ_1 regularization dominates ($\mu_1 = \nu_1 = 0.1, \mu_2 = \nu_2 = 0$), Algorithm 2 is about 20 times faster than Algorithm 1.

5.3 Convergence Behavior

We ran Algorithm 2 up to 600 iterations for the unregularized model ($\mu_1 = \nu_1 = \mu_2 = \nu_2 = 0$), and 10^4 iterations for the regularized model where we set $\mu_1 = \nu_1 = 0.01$ and $\mu_2 = \nu_2 = 0.5$. For both models, $r = 1$ was used. The last iterate was used as \mathbf{W}^* . The dataset is described in Section 6.1.1.

Figure 3 shows the convergence behavior of Algorithm 2 for solving (4) with different regularization terms. From the figure, we see that our algorithm converges pretty fast and the difference $\|\mathbf{W}^k - \mathbf{W}^*\|_F$ appears to decrease linearly at first and superlinearly eventually.

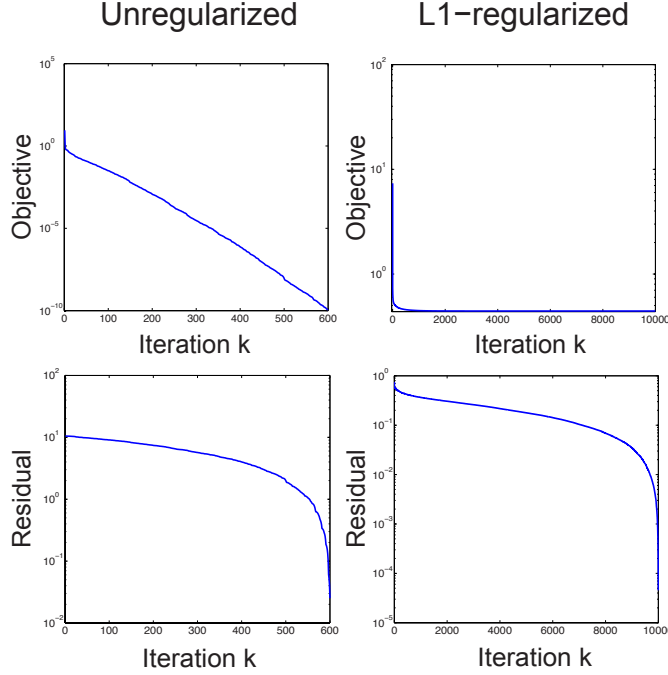


Figure 3: Convergence behavior for solving (4) using Algorithm 2. Top panel plots the objective function as a function of iteration. Bottom panel plots the residual $\|\mathbf{W}^k - \mathbf{W}^*\|_F$ as a function of iteration.

6 Applications

We apply sparse bilinear logistic regression to several real-world applications and compare its generalization performance with logistic regression, sparse logistic regression and bilinear logistic regression. We also extend the sparse bilinear logistic regression from the binary case to multi-class case in several experiments.

6.1 Brain Computer Interface

6.1.1 Binary Case

We tested the classification performance of sparse bilinear logistic regression (4) on an EEG dataset with binary labels. We used the EEG dataset IVb from BCI competition III². Dataset IVb concerns a motor imagery classification task. The 118 channel EEG was recorded from a healthy subject sitting in a comfortable chair with arms resting on armrests. Visual cues (letter presentation) were shown for 3.5 seconds, during which the subject performed: left hand, right foot, or tongue. The data was sampled at 100 Hz, and

²<http://www.bbc.de/competition/iii/>

Table 2: Classification performance for the BCI EEG dataset.

Models	Prediction Accuracy
Logistic Regression	0.75
Sparse Logistic Regression	0.76
Bilinear Logistic Regression	0.84
Sparse Bilinear Logistic Regression	0.89

the cues of “left hand” and “right foot” were marked in the training data. We chose all the 210 marked data points for test and downsampled each point to have 100 temporal slices, namely, $s = 118$, $t = 100$ in this test.

In (4), there are five parameters $\mu_1, \mu_2, \nu_1, \nu_2$ and r to be tuned. Leave-one-out cross validation was performed on the training dataset to tune these data. First, we fixed $\mu_1 = \mu_2 = \nu_1 = \nu_2 = 0$ (i.e., unregularized) and tuned r . Then, we fixed r to the previously tuned one ($r = 1$ in this test) and selected the best $(\mu_1, \mu_2, \nu_1, \nu_2)$ from a $6 \times 5 \times 6 \times 5$ grid.

Table 2 shows the prediction accuracy on the testing dataset. We used the ROC analysis to compute the Az value (area under ROC curve) for both the unregularized model and the regularized model, where the best hyperparameters for the regularized model are tuned on the validation dataset using cross validation. We compared (sparse) logistic regression with (sparse) bilinear logistic regression. We solved the ℓ_1 -regularized logistic regression using FISTA [2]. We observed that bilinear logistic regression gives much better predictions than logistic regression. In addition, sparse bilinear logistic regression performs better than the unregularized bilinear logistic regression.

6.1.2 Multi-class Case

Table 3: Classification performance for the multi-class EEG dataset.

Models	Prediction Accuracy
Logistic Regression	0.54
Sparse Logistic Regression	0.54
Bilinear Logistic Regression	0.55
Sparse Bilinear Logistic Regression	0.65

We further extended our sparse bilinear logistic regression to the multi-class case using one-versus-all method. The EEG dataset in this experiment was based on a cognitive

experiment where the subject view images of three categories and tried to make a decision about the category [25]. The data was recorded at 2048 Hz using a 64-channel EEG cap. We downsampled this data to 100 Hz.

Table 3 shows classification performance for the multi-class classification. Consistently for all the three stimuli, bilinear logistic regression outperforms logistic regression, and sparse bilinear logistic regression further improves the generalization performance by introducing sparsity.

6.2 Separating Style and Content

As mentioned earlier, one benefit of the bilinear model is to separate style and content. In order to exploit this property, we classified images with various camera viewpoints.

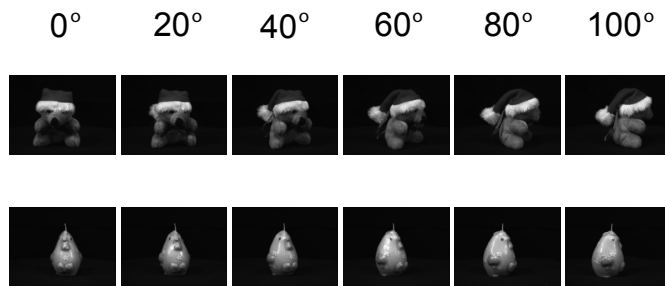


Figure 4: Sample images with various camera viewpoints.

We used the Amsterdam Library of Object Images,³ where the frontal camera was used to record 72 viewpoints of the objects by rotating the object in the plane at 5° resolution from 0° to 355°. Figure 4 shows some sample images with various camera viewpoints.

Table 4: Classification performance for images with various camera viewpoints.

Models	Prediction Accuracy
Logistic Regression	0.86
Sparse Logistic Regression	0.86
Bilinear Logistic Regression	0.94
Sparse Bilinear Logistic Regression	1.00

Table 4 shows the comparison between (sparse) logistic regression and (sparse) bilinear logistic regression. We observe a significant improvement using the bilinear model, and sparse bilinear logistic regression achieves the best generalization performance.

³<http://staff.science.uva.nl/~aloi/>

6.3 Visual Recognition of Videos

We used sparse bilinear logistic regression to videos [32], in the context of visual recognition for UCF sports action dataset.⁴ Since the size of the original video is big, we reduced the dimensionality of feature space by extracting histograms based on scale-invariant feature transform (SIFT) descriptors [26] for each frame.

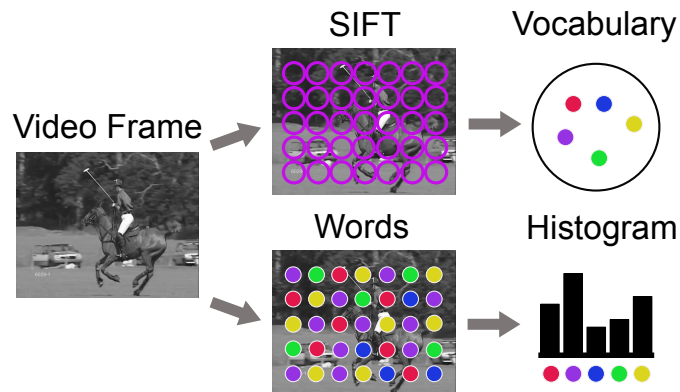


Figure 5: Illustration of building SIFT histogram features.

Figure 5 illustrates such a procedure. We first built a vocabulary for the codebook assuming 100 words, using k-mean clustering based on all the SIFT descriptors across frames for all the videos. We then constructed histograms for each frame according to the codebook. A tiling technique was used to improve the performance. This procedure reduced the feature space to $s = 400$ and $t = 55$.

We focused on five classes of sports action and we used the following abbreviations: Diving (Diving-Side), Riding (Riding-Horse), Run (Run-Side), Swing (Swing-Sideangle), Walk (Walk-Front). We picked 6 videos out of each class, and used 6-fold cross validation to test discrimination accuracy in the context of transfer learning.

Table 5: Classification performance for the UCF sports action video dataset.

Models	Prediction Accuracy
Logistic Regression	0.70
Sparse Logistic Regression	0.70
Bilinear Logistic Regression	0.73
Sparse Bilinear Logistic Regression	0.77

⁴http://crcv.ucf.edu/data/UCF_Sports_Action.php

Table 5 shows the classification performance for (sparse) logistic regression and (sparse) bilinear logistic regression. In overall, sparse bilinear logistic regression achieves the best classification performance.

7 Discussion

We proposed sparse bilinear logistic regression, and developed an efficient numerical algorithm using the block coordinate proximal descent method. Theoretical analysis revealed its global convergence as well as convergence rate. We demonstrated its generalization performance on several real-world applications.

7.1 Dimensionality Reduction and Classification

It should be noted that bilinear logistic regression performs dimensionality reduction and classification within the same framework. Traditionally in order to combat the curse of dimensionality, dimension reduction techniques such as principle component analysis (PCA) and independent component analysis (ICA) were commonly used as a preprocessing step before carrying out classification. Instead of a two-step processing, bilinear logistic regression carries out dimension reduction and classification using one optimization problem.

Sparse bilinear logistic regression further fuses the benefits of sparse logistic regression and bilinear logistic regression into the same framework. Sparsity overcomes the ambiguity intrinsic to the bilinear model, which is critical to the quality of solution. Sparsity leads to feature selection in both spatial and temporal domains. More importantly, sparsity improves the generalization performance of the classifier, which is intimately related to the logarithmic sample complexity [29]. We demonstrated such an improvement using a range of numerical experiments. However, it remains a challenging problem to carry out rigorous statistical analysis based on the minimax theory due to the bi-convex nature.

7.2 Bi-convexity: More Gain than Pain

Bilinear model introduces bi-convexity into the objective function, however, it should be noted that the resulting decision boundary is still linear. Recall the objective function for sparse bilinear logistic regression is the following

$$\min_{\mathbf{U}, \mathbf{V}, b} \frac{1}{n} \sum_{i=1}^n \log \left(1 + \exp[-y_i(\text{tr}(\mathbf{U}^\top \mathbf{X}_i \mathbf{V}) + b)] \right) + r_1(\mathbf{U}) + r_2(\mathbf{V}).$$

The estimated spatial factor $\mathbf{U} \in \mathbb{R}^{s \times r}$ and temporal factor $\mathbf{V} \in \mathbb{R}^{r \times t}$ essentially forms a low-rank weight matrix $\mathbf{W} \in \mathbb{R}^{s \times t}$, $\mathbf{W} = \mathbf{U}\mathbf{V}^\top$. Hence the decision boundary can be

written as

$$\langle \text{diag}(\mathbf{W} \otimes \mathbf{1}), x \rangle + b = 0.$$

With such an interpretation, one can also reformulate the objective function of the sparse bilinear logistic regression as

$$\min_{\mathbf{W}, \mathbf{U}, \mathbf{V}, b} \frac{1}{n} \sum_{i=1}^n \log(1 + \exp[-y_i(\text{tr}(\mathbf{W} \otimes \mathbf{X}_i) + b)]) + \lambda \|\mathbf{W}\|_* + r_1(\mathbf{U}) + r_2(\mathbf{V}), \quad \mathbf{W} = \mathbf{U}\mathbf{V}^\top.$$

We see that the bilinear logistic regression has an equivalent convex formulation by minimizing the nuclear norm of \mathbf{W} . However, due to the benefits of sparsity (as discussed in section 2.3.2), it becomes critical to have a bilinear factorization and impose sparsity promoting priors on the spatial and temporal factors.

As much as the difficulty of statistical analysis posed by bi-convexity, our numerical algorithm for solving sparse bilinear logistic regression is extremely efficient and has a guarantee for global convergence. As we demonstrated empirically on a variety of classification tasks, sparse bilinear logistic regression provides an avenue to boost generalization performance.

7.3 Multinomial Generalization

The binomial sparse bilinear logistic regression can be further generalized to the multinomial case. We assume each sample $\{\mathbf{x}_i\}$ to belong to $(m+1)$ classes and label $y_i \in \{1, 2, \dots, m+1\}$ and seek $(m+1)$ hyperplanes $\{\mathbf{x} : \mathbf{w}_c^\top \mathbf{x} + b_c = 0\}_{c=1}^{m+1}$ to separate these samples. According to the logistic model, the conditional probability for y_i based on sample \mathbf{x}_i is

$$P(y_i = c | \mathbf{x}_i, \mathbf{w}, \mathbf{b}) = \frac{\exp[\mathbf{w}_c^\top \mathbf{x}_i + b_c]}{\sum_{j=1}^{m+1} \exp[\mathbf{w}_j^\top \mathbf{x}_i + b_j]}, \quad c = 1, \dots, m+1. \quad (32)$$

Because of the normalization condition $\sum_{c=1}^{m+1} P(y_i = c | \mathbf{x}_i, \mathbf{w}, \mathbf{b}) = 1$, one (\mathbf{w}_c, b_c) needs not be estimated. Without loss of generality, we set $(\mathbf{w}_{m+1}, b_{m+1})$ to *zero*. Let $y_{ic} = 1$ if $y_i = c$ and $y_{ic} = 0$ otherwise. Then (32) becomes

$$P(y_i | \mathbf{x}_i, \mathbf{w}, \mathbf{b}) = \frac{\exp[\sum_{c=1}^m y_{ic}(\mathbf{w}_c^\top \mathbf{x}_i + b_c)]}{1 + \sum_{c=1}^m \exp[\mathbf{w}_c^\top \mathbf{x}_i + b_c]}. \quad (33)$$

The average negative log-likelihood function is

$$\begin{aligned} \mathcal{L}(\mathbf{w}, \mathbf{b}) &= -\frac{1}{n} \sum_{i=1}^n \log P(y_i | \mathbf{x}_i, \mathbf{w}, \mathbf{b}) \\ &= \frac{1}{n} \sum_{i=1}^n \left(\log \left(1 + \sum_{c=1}^m \exp[\mathbf{w}_c^\top \mathbf{x}_i + b_c] \right) - \sum_{c=1}^m y_{ic}(\mathbf{w}_c^\top \mathbf{x}_i + b_c) \right) \end{aligned}$$

To perform MLE for (\mathbf{w}, \mathbf{b}) , one can minimize $\mathcal{L}(\mathbf{w}, \mathbf{b})$. Under the above setting, where each sample is a matrix and each weight \mathbf{w}_c has the form of $\mathbf{U}_c \mathbf{V}_c^\top$, the loss function becomes

$$\mathcal{L}(\mathbf{U}, \mathbf{V}, \mathbf{b}) = \frac{1}{n} \sum_{i=1}^n \left(\log \left(1 + \sum_{c=1}^m \exp[\text{tr}(\mathbf{U}_c^\top \mathbf{X}_i \mathbf{V}_c) + b_c] \right) - \sum_{c=1}^m y_{ic} (\text{tr}(\mathbf{U}_c^\top \mathbf{X}_i \mathbf{V}_c) + b_c) \right).$$

The multinomial sparse bilinear logistic regression takes the following variational formulation

$$\min_{\mathbf{U}, \mathbf{V}, \mathbf{b}} \mathcal{L}(\mathbf{U}, \mathbf{V}, \mathbf{b}) + R_1(\mathbf{U}) + R_2(\mathbf{V}), \quad (34)$$

where $\mathbf{U} = (\mathbf{U}_1, \dots, \mathbf{U}_m)$, $\mathbf{V} = (\mathbf{V}_1, \dots, \mathbf{V}_m)$ with $\mathbf{U}_c \in \mathbb{R}^{S \times K}$ and $\mathbf{V}_c \in \mathbb{R}^{T \times K}$ for each class c , and R_1 and R_2 are used to promote priori structures on \mathbf{U} and \mathbf{V} , respectively.

References

- [1] H. Attouch and J. Bolte. On the convergence of the proximal algorithm for nonsmooth functions involving analytic features. *Mathematical Programming*, 116:5–16, 2009.
- [2] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.
- [3] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2007.
- [4] J. Bochnak, M. Coste, and M. F. Roy. *Real Algebraic Geometry*, volume 36. Springer Verlag, 1998.
- [5] J. Bolte, A. Daniilidis, and A. Lewis. The Lojasiewicz inequality for nonsmooth sub-analytic functions with applications to subgradient dynamical systems. *SIAM Journal on Optimization*, 17(4):1205–1223, 2007.
- [6] S. Boyd, S.-J. Kim, L. Vandenberghe, and A. Hassibi. A tutorial on geometric programming. *Optimization and Engineering*, 8(1):67–127, 2007.
- [7] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.
- [8] M. Dyrholm, C. Chistoforou, L. C. Parra, and P. Kaelbling. Bilinear discriminant component analysis. *Journal of Machine Learning Research*, 8:1007–1021, 2007.
- [9] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *Annals of Statistics*, 32(2):407–499, 2004.

- [10] S. Eyheramendy, A. Genkin, W. Ju, D. Lewis, and D. Madigan. Sparse bayesian classifiers for text categorization. Technical report, Journal of Intelligence Community Research and Development, 2003.
- [11] M. Figueiredo. Adaptive sparseness for supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25:1150–1159, 2003.
- [12] M. Figueiredo and A. Jain. Bayesian learning of sparse classifiers. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 35–41, 2001.
- [13] A. Genkin, D. D. Lewis, and D. Madigan. Large-scale bayesian logistic regression for text categorization. *Technometrics*, 49(3):291–304, 2007.
- [14] J. Goodman. Exponential priors for maximum entropy models. In *Proc. Annual Meetings of the Association for Computational Linguistics (ACL)*, pages 305–312, 2004.
- [15] R. A. Harshman. Foundations of the PARAFAC procedure: models and conditions for an” explanatory” multi-modal factor analysis. *UCLA Working Papers in Phonetics*, 16(1):1–84, 1970.
- [16] D. W. Hosmer and S. Lemeshow. *Applied Logistic Regression*. Probability and Statistics. Wiley, 2nd edition, 2000.
- [17] K. Koh, S.-J. Kim, and S. Boyd. An interior-point method for large-scale ℓ_1 -regularized logistic regression. *Journal of Machine Learning Research*, 8:1519–1555, 2007.
- [18] B. Krishnapuram, L. Carin, and M. Figueiredo. Sparse multinomial logistic regression: fast algorithms and generalization bounds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(6):957–968, 2005.
- [19] K. Kurdyka. On gradients of functions definable in o-minimal structures. *Annales de l’Institut Fourier*, 48(3):769–784, 1998.
- [20] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- [21] S. Lee, H. Lee, P. Abbeel, and A. Ng. Efficient ℓ_1 -regularized logistic regression. In *Proc. National Conference on Artificial Intelligence (AAAI)*, 2006.
- [22] J. G. Liao and K. V. Chin. Logistic regression for disease classification using microarray data: model selection in a large p and small n cas. *Bioinformatics*, 23(15):1945–51, 2007.

- [23] S. Łojasiewicz. Sur la géométrie semi-et sous-analytique. *Annales de L’Institut Fourier (Grenoble)*, 43(5):1575–1595, 1993.
- [24] J. Lohhorst. The lasso and generalised linear models. Technical report, Honors Project, Department of Statistics, University of Adelaide, South Australia, Australia, 1999.
- [25] B. Lou, J. M. Walz, J. V. Shi, and P. Sajda. Learning EEG components for discriminating multi-class perceptual decisions. In *Proc. IEEE Conference on Neural Engineering (NER)*, pages 675–678, 2011.
- [26] D. G. Lowe. Object recognition from local scale-invariant features. In *Proc. International Conference on Computer Vision (ICCV)*, volume 2, pages 1150–1157, 1999.
- [27] Z. Q. Luo and P. Tseng. Error bounds and convergence analysis of feasible descent methods: A general approach. *Annals of Operations Research*, 46(1):157–178, 1993.
- [28] D. Madigan, A. Genkin, D. Lewis, and D. Fradkin. Bayesian multinomial logistic regression for author identification. In *Proc. Maxent Conference*, pages 509–516, 2005.
- [29] A. Ng. Feature selection, l_1 vs l_2 regularization, and rotational invariance. In *Proc. International Conference on Machine Learning (ICML)*, pages 78–85. ACM Press, New York, 2004.
- [30] L. C. Parra, C. D. Spence, A. D. Gerson, and P. Sajda. Recipes for the linear analysis of eeg. *Neuroimage*, 28(2):326–341, 2005.
- [31] S. Perkins and J. Theiler. Online feature selection using grafting. In *Proc. International Conference on Machine Learning (ICML)*, pages 592–599, 2003.
- [32] H. Pirsiavash, D. Ramanan, and C. Fowlkes. Bilinear classifiers for visual recognition. In *Advances in Neural Information Processing Systems (NIPS)*, 2009.
- [33] R. T. Rockafellar and R. J. B. Wets. *Variational Analysis*, volume 317. Springer Verlag, 1998.
- [34] V. Roth. The generalized lasso. *IEEE Transactions on Neural Networks*, pages 16–28, 2004.
- [35] J. Shi, W. Yin, S. Osher, and P. Sajda. A fast hybrid algorithm for large-scale ℓ_1 -regularized logistic regression. *Journal of Machine Learning Research*, pages 581–609, 2010.
- [36] J. B. Tenenbaum and W. T. Freeman. Separating style and content with bilinear models. *Neural Computation*, 12(6):1247–1283, 2000.

- [37] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58(1):267–288, 1996.
- [38] P. Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of Optimization Theory and Applications*, 109:475–494, 2001.
- [39] P. Tseng and S. Yun. A coordinate gradient descent method for nonsmooth separable minimization. *Mathematical Programming*, 117:387–423, 2009.
- [40] Y. Tsuruoka, J. McNaught, J. Tsujii, and S. Ananiadou. Learning string similarity measures for gene/protein name dictionary look-up using logistic regression. *Bioinformatics*, 23(20):2768–74, 2007.
- [41] J. Vidal. Real-time detection of brain events in EEG. *Proceedings of the IEEE*, 65(5):633–641, 1977.
- [42] Y. Xu and W. Yin. A block coordinate descent method for regularized multi-convex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on Imaging Science*, 6(3):1758–1789, 2013.
- [43] J. Zhu and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society, Series B*, 67(2):301–320, 2005.
- [44] J. Zhu, S. Rosset, T. Hastie, and R. Tibshirani. 1-norm support vector machines. In *Advances in Neural Information Processing Systems (NIPS)*, volume 16, pages 49–56, 2004.